Preprints.org

Article

Not peer-reviewed version

Sub-Millijoule Intrusion Detection on a Commodity MCU Neural Processing Unit: A Four-Dataset Deployment Study

Hsiu-Chi Tsai *

Posted Date: 16 April 2026

doi: 10.20944/preprints202603.0817.v3

Keywords: intrusion detection; spiking neural network; neural processing unit; INT8 quantization; edge AI; STM32N6; Neural-ART

Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Sub-Millijoule Intrusion Detection on a Commodity MCU Neural Processing Unit: A Four-Dataset Deployment Study

Hsiu-Chi Tsai

Department of Computer Science, National Yang Ming Chiao Tung University, Hsinchu, Taiwan; hctsai1006@cs.nctu.edu.tw

Abstract

We deploy an intrusion detection classifier on the STM32N6570-DK, a Cortex-M55 MCU with the Neural-ART NPU. Using the approximate T=1 SNN–INT8 ANN equivalence, we compile a lightweight MLP to the NPU and evaluate four datasets: NSL-KDD (5-class), UNSW-NB15 (10-class), CICIDS2017 (15-class), and IoT-23 (5-class). Results are reported as mean±std over multi-seed runs (5–20 seeds), with paired Wilcoxon signed-rank tests and Holm–Bonferroni correction. Across all datasets, INT8 NPU inference runs in **0.29–0.46 ms** (2.7–4.2× faster than the same model on Cortex-M55 CPU), with estimated energy **44–69 µJ/inference** and Flash 105–138 KB. Compared with recent MCU-class deployments on STM32F7 (31 ms, 7.86 mJ) and Raspberry Pi 3B+ (27 ms), our path delivers 59–107× lower latency; the estimated energy envelope implies 114–179× lower energy than STM32F7. QCFS and ReLU are statistically indistinguishable on all four datasets ($p \geq 0.227$), supporting practical T=1 near-equivalence under commodity MCU deployment constraints. Energy is estimated from AN5946 rather than direct on-board measurement, and UNSW shows greater quantization fragility than NSL-KDD.

Keywords: intrusion detection; spiking neural network; neural processing unit; INT8 quantization; edge AI; STM32N6; Neural-ART

1. Introduction

Deep-learning-based intrusion detection at the IoT edge must operate under tight power and latency budgets. Recent MCU-class deployments achieve promising accuracy but at high energy cost: Chehade et al. reported 31 ms and 7.86 mJ per inference on an STM32F746G (Cortex-M7, no NPU) [1], and Diab et al. required 27 ms on a Raspberry Pi 3B+ [2]. At these latencies, CPU-only inference blocks all concurrent RTOS tasks for tens of milliseconds, making real-time packet processing difficult.

Commodity MCUs are beginning to integrate neural processing units (NPUs) that execute INT8 matrix operations at hundreds of GOPS while freeing the CPU for other tasks. A line of theoretical work [3–5] has shown that single-timestep (T=1) SNN inference is approximately equivalent to INT8 quantized ANN inference: when the membrane potential is initialized to zero, the threshold-and-fire mechanism reduces to a ReLU-like clamp. Under the stated conversion and quantization assumptions, NPUs that execute INT8 `Gemm+Relu` can approximate single-timestep SNN-equivalent behavior.

We validate this on the STM32N6570-DK (ARM Cortex-M55 @ 800 MHz, Neural-ART NPU, 600 GOPS INT8). We focus on classifier inference latency, energy, and memory footprint on pre-extracted flow-level features; the full IDS pipeline (packet capture, flow aggregation) is left to future work.

Our contributions:

- Following a systematic literature search protocol (Supplementary File S1: `novelty_search_protocol.md`), the first publicly documented IDS classifier deployment on a Cortex-M class MCU paired with a

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

general-purpose NPU (Neural-ART), achieving 0.29–0.46 ms inference at an estimated 44–69 µJ per inference.
- A four-dataset multi-seed study (5–20 seeds per arm, see Table 3 caption) with paired Wilcoxon signed-rank tests under Holm–Bonferroni family-wise error control, covering NSL-KDD, UNSW-NB15, CICIDS2017, and IoT-23.
- A non-MLP TinyCNN baseline (Conv2D 1×3) under the same Neural-ART operator constraints, enabling same-hardware comparison.
- Empirical validation of practical T=1 SNN–ANN approximation on commercial NPU silicon across four datasets, with 99% prediction agreement between FP32 and INT8 models on NSL-KDD.
- A QCFS sweep ($L \in \{2, 4, 8, 16\}$) justifying L=4 and quantifying `Floor`-triggered CPU fallback (+17.6% latency).

2. Related Work

Table 1 compares IDS deployments on physical hardware. Ngo et al. [6] deployed a 1,360-parameter MLP on the MAX78000, an AI-specialized MCU with a fixed CNN accelerator, achieving 98.57% on UNSW-NB15 (binary) at 18 mW. Zahm et al. [7] used the BrainChip Akida AKD1000 neuromorphic processor (98.4%, ∼1 W). Both use purpose-built AI/neuromorphic hardware. Chehade et al. [1] deployed a 1D-CNN on the STM32F746G (Cortex-M7, no NPU), achieving 96.59% accuracy on ISCX VPN-nonVPN but requiring 31 ms and 7.86 mJ per inference. Diab et al. [2] deployed LightGBM/CNN on Raspberry Pi 3B+ (27–300 ms, 250–275 mW). Farooq et al. [8] report 1.16 G inferences/sec on a Xilinx FPGA, but evaluate on the Edge-IIoT dataset and operate in a different hardware class (∼10 W FPGA vs. ∼150 mW MCU), so latency and energy are not directly comparable. Our work targets a general-purpose MCU with an attached programmable NPU, achieving two orders of magnitude lower energy than CPU-only MCU deployments. Following the systematic literature search protocol documented in Supplementary File S1 (`novelty_search_protocol.md`; 5 databases, 8 query variants, ≈320 records inspected), we find no prior publicly documented IDS classifier deployed on a commodity ARM Cortex-M-class MCU paired with a general-purpose neural accelerator.

GPU-based SNN-IDS systems [9–11] report 94–99% on NSL-KDD/UNSW-NB15, but mostly in binary settings and without MCU-class deployment paths. Our multi-class formulation (5/10/15/5-class across NSL-KDD, UNSW-NB15, CICIDS2017, and IoT-23) and hardware deployment target a different design point.

Table 1. Hardware-deployed IDS comparison. "Class" denotes hardware class: **MCU** = commodity microcontroller, **AI-MCU** = AI-specialized MCU with fixed CNN engine, **ASIC** = neuromorphic ASIC, **SBC** = single-board computer, **FPGA** = field- programmable gate array.

Work	Platform	Class	Task	Acc.	Lat.	Energy	NPU
Ngo [6]	MAX78000	AI-MCU	bin.	98.6%	—	18 mW*	CNN eng.
Zahm [7]	Akida	ASIC	multi	98.4%	—	∼1 W*	Neurom.
Chehade [1]	STM32F7	MCU	multi	96.6%	31 ms	7.86 mJ	None
Diab [2]	RPi 3B+	SBC	multi	95.3%	27 ms	∼6.75 mJ‡	None
Farooq [8]§	Xilinx FPGA	FPGA	bin.	—	<1 µs	—	—
This work	**STM32N6**	**MCU**	**5/10/15/5**	**78.6/64.7/91.9/75.6%†**	**0.29 ms**	**44 µJ**	**NPU**

*Power, not energy; latency not reported. †NSL-KDD 5-class / UNSW-NB15 10-class / CICIDS2017 15-class / IoT-23 5-class. ‡Estimated from reported power and latency. §Edge-IIoT dataset, different hardware class (∼10 W FPGA).

3. System Design

3.1. T=1 SNN–ANN Equivalence

A Leaky Integrate-and-Fire (LIF) neuron updates its membrane potential $V[t] = \beta \cdot V[t-1] + \mathbf{W}\mathbf{x}[t]$ and fires when $V[t] > \theta$. At T=1 with $V[0]$=0, the leak term vanishes and firing reduces to $\Theta(\mathbf{W}\mathbf{x} - \theta)$, which is equivalent to a quantized ReLU. Bu et al. [3] formalized this via QCFS; Jiang

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

et al. [4] and Bu et al. [5] further showed that INT8 post-training quantization produces a discretization grid that closely approximates the threshold-and-fire operation. In practice, under this QCFS/INT8 conversion setting, NPUs that execute INT8 `Gemm+Relu` can approximate the same decision function as the $T{=}1$ classifier.

3.2. Neural-ART NPU

The STM32N6570-DK pairs an ARM Cortex-M55 at 800 MHz with the Neural-ART NPU rated at 600 GOPS for INT8 and 3 TOPS/W efficiency [12]. The NPU supports `Conv`, `Gemm`, `Relu`, `Add`, `Clip`, and `MaxPool` in INT8; unsupported operators fall back to the Cortex-M55 CPU with automatic cache-maintenance overhead at each NPU↔CPU handoff. The board includes 4.2 MB internal SRAM and 128 MB external NOR Flash.

3.3. Model Architecture

We use a four-layer MLP with BatchNorm and ReLU: $\texttt{Lin}(d{\to}256) \to \texttt{BN} \to \texttt{ReLU} \to \texttt{Lin}(256{\to}256) \to \texttt{BN} \to \texttt{ReLU} \to \texttt{Lin}(256{\to}128) \to \texttt{BN} \to \texttt{ReLU} \to \texttt{Lin}(128{\to}C)$, where $(d, C) \in \{(41, 5), (34, 10), (69, 15), (13, 5)\}$ for NSL-KDD, UNSW-NB15, CICIDS2017, and IoT-23 respectively. BatchNorm is folded into the preceding Linear at ONNX export, yielding a graph of `Gemm+Relu` operators only (103–119 K parameters, dominated by the $256{\times}256$ hidden layer). The architecture is deliberately shallow: deeper models could improve minority-class recall but risk introducing operators outside the NPU's supported set. All hidden dimensions are powers of two (256, 128), avoiding a known Neural-ART compiler issue with prime-number channel counts [12].

3.4. Datasets and Training

NSL-KDD [13]: 125,973 train / 22,544 test, 41 features, 5-class. **UNSW-NB15** [14]: 175,341 train / 82,332 test, 34 features, 10-class. **CICIDS2017** [15]: ≈2.26 M train / 566 K test, 78 raw features (69 after removing all-constant columns), 15-class. We use the HuggingFace cleaned version, which corrects the TCP-splitting and label-alignment issues documented by Engelen et al. [16]. **IoT-23** [17]: ≈4.84 M train / 1.21 M test, 13 features, 5-class (Benign, C&C, DDoS, Okiru, PortScan). This dataset provides contemporary IoT botnet traffic from 2018–2019 captures. Training: Adam (lr$=10^{-3}$), cosine annealing, 40–80 epochs, inverse-frequency class weighting. ReLU arms use $n{=}20$ seeds on NSL-KDD and UNSW-NB15, and $n{=}10$ on CICIDS2017 and IoT-23; QCFS arms use $n{=}10$ on UNSW-NB15 and $n{=}5$ on CICIDS2017 and IoT-23 due to training-set size. Three categorical features are label-encoded; remaining features are z-score normalized using training-set statistics. FP32 models are quantized to INT8 via ONNX Runtime static PTQ (MinMax, per-tensor, 1,000 calibration samples). A 24-configuration ablation (Section 4.3) confirms that the choice of calibration method, granularity, and sample size changes accuracy by at most 0.82 pp on NSL-KDD.

Dataset caveats. NSL-KDD is derived from 1998 DARPA traffic and does not reflect modern IoT/5G traffic distributions; we retain it for benchmark continuity with the MCU-IDS literature (HH-NIDS [6], Chehade [1], Diab [2] all report NSL-KDD-derived results). CICIDS2017's original release contained TCP splitting bugs, duplicate flows, and mislabeled port scans [16]; the cleaned HuggingFace version we use mitigates but does not eliminate these issues. We report per-class recall in Section 4 to expose label-noise effects.

Fair-comparison caveats. The MCU-IDS literature uses heterogeneous preprocessing: Ngo et al. [6] apply SMOTE oversampling and report binary accuracy; Chehade et al. [1] use ISCX VPN-nonVPN (traffic classification, not intrusion detection); Farooq et al. [8] use Edge-IIoT at gigabit line rate on FPGA. We deliberately report multi-class accuracy with inverse-frequency class weighting (no oversampling) on four intrusion-detection benchmarks, and confine our head-to-head numerical comparison to platforms in the same hardware class (commodity MCU).

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

3.5. NPU Compilation and Operator Mapping

Models are compiled for the STM32N6570-DK using ST Edge AI Developer Cloud v4.0.0 (`atonn` v1.1.3). Table 2 shows the operator-level mapping. All `Gemm+Relu` layers map to NPU hardware epochs. The `QuantizeLinear`/`DequantizeLinear` operators at the model boundary may require CPU epochs depending on the input dimension.

Table 2. Operator mapping to NPU epochs. ReLU INT8 achieves near-100% NPU utilization; QCFS INT8 forces CPU fallback at every `Floor`.

ONNX Operator	NPU Support	Epoch Type	Notes
`Gemm` (INT8)	✓	HW	Fused weight + bias
`Relu`	✓	HW	Fused with preceding Gemm
`Clip`	✓	HW	Used by QCFS
`Floor`	✗	SW (float)	CPU fallback, +0.08 ms
`QuantizeLinear`	✓	HW/Hyb	At model boundary
`DequantizeLinear`	✓	HW/Hyb	At model boundary

For the ReLU INT8 path: NSL-KDD compiles to 5 HW + 1 Hyb + 2 SW epochs (0.46 ms); UNSW-NB15 compiles to 4 HW + 0 Hyb + 0 SW epochs (0.29 ms, 100% NPU). The QCFS INT8 path compiles to 13 HW + 1 Hyb + 14 SW epochs (0.54 ms) because each `Floor` triggers a CPU round-trip via `Dequant`→`Floor`→`Quant`. The CICIDS2017 and IoT-23 INT8 ONNX graphs share the same `Gemm+Relu` structure and compile to 4 HW + 0 SW epochs (100% NPU), achieving 0.42 ms and 0.38 ms respectively.

4. Experiments

4.1. Classification Results

Table 3 reports multi-seed results. On NSL-KDD, ReLU MLP and QCFS MLP are statistically indistinguishable (p=0.227, Table 7), which provides evidence of non-rejection of T=1 SNN–ANN approximation, since the QCFS-trained network behaves like the ReLU model once BatchNorm is folded and the graph is quantized to INT8. The same non-rejection pattern appears on UNSW-NB15, CICIDS2017, and IoT-23 (Table 3, QCFS rows), where the Wilcoxon signed-rank test against ReLU yields p=0.846, p=0.312, and p=0.438 respectively; we report them in Table 7. Taken together, these results extend the NSL-KDD finding to four heterogeneous datasets and support cross-dataset evidence for practical T=1 approximation on a commodity MCU IDS workload, not formal statistical equivalence. TinyCNN (Conv2D 1×3, 3.7K params) exhibits a large but underpowered effect on NSL-KDD (d_z=−0.97, p_{adj}=0.055) where it trends higher than the MLP. On UNSW-NB15 the MLP *significantly* outperforms TinyCNN (p=0.002, d_z=2.06). On CICIDS2017 the MLP again leads with a large effect (d_z=+1.32, all five seeds direction-consistent), but at n=5 the Wilcoxon test hits the two-sided floor (p=0.063), so we do not claim significance. Neither topology dominates across datasets, which is itself a deployment finding: an RTOS scheduler could host both models (combined Flash <300 KB) and dispatch by traffic feature. On UNSW-NB15 (10-class), Random Forest achieves higher overall accuracy. In our tested toolchain/backend configuration, it cannot be compiled to the NPU path: ST Edge AI Core rejects `TreeEnsembleClassifier` with "NOT IMPLEMENTED." While ST documentation lists ONNX-ML tree operators in the toolbox support, our tested path did not provide a Neural-ART deployment route for this model class [18]. We therefore report MLP/CNN deployment results for the Neural-ART path. The accuracy gap reflects the difficulty of 10-class classification with extreme imbalance (Worms: 44 samples), and is consistent with a dataset-limited regime rather than a clear modeling limitation specific to NPU deployment.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

Table 3. Test accuracy (%, mean±std, 5–20 seeds per arm; NSL-KDD ReLU/QCFS n=20, UNSW ReLU n=20/QCFS n=10, CICIDS ReLU n=10/QCFS n=5, IoT-23 ReLU n=10/QCFS n=5). MLP and TinyCNN are the only NPU-eligible models. †TinyCNN uses Conv2D 1×3 kernels, the only non-MLP topology that fits within Neural-ART's operator set.

Dataset	Model	Overall	Macro F1	NPU?
NSL-KDD	ReLU MLP	78.57±1.28	58.91±2.80	✓
	QCFS MLP	78.14±1.08	58.28±2.70	Partial
	TinyCNN†	**80.32±0.48**	**60.69±0.77**	✓
	Rand. Forest	73.84±0.19	47.13±0.33	✗
UNSW-NB15	ReLU MLP	64.67±0.55	40.18±1.02	✓
	QCFS MLP	64.67±0.49	39.94±0.72	Partial
	TinyCNN†	63.28±0.23	38.12±0.47	✓
	Rand. Forest	**69.46±0.10**	**48.63±0.38**	✗
CICIDS2017	ReLU MLP	91.89±1.21	56.35±2.80	✓
	QCFS MLP	90.99±0.48	57.65±2.38	Partial
	TinyCNN†	90.57±0.13	56.74±0.13	✓
IoT-23	ReLU MLP	75.59±2.71	66.41±1.50	✓
	QCFS MLP	77.65±1.18	67.40±0.52	Partial

Per-class analysis on NSL-KDD reveals that DoS (F1=86.9%) and Normal (82.4%) are well-classified, while R2L suffers from low recall (25.3%); 63.2% of R2L samples are misclassified as Normal. U2R (200 test samples) scatters across Probe (45.4%) and Normal (29.2%). On UNSW-NB15, Generic achieves F1=97.9%, but minority classes collapse: Worms F1=9.1%, Analysis F1=2.9%. These patterns are characteristic of extreme class imbalance, not NPU-induced degradation on NSL-KDD; FP32 and INT8 models agree on 99% of predictions.

Fig. 1 visualizes these patterns. The class imbalance is severe: R2L has only 995 training samples (0.8% of the set) yet 2,754 test samples (12.2%); U2R has 52 training samples.

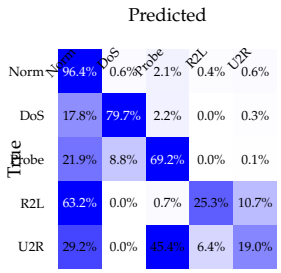

Figure 1. NSL-KDD confusion matrix (%, 20-seed mean, ReLU MLP). R2L→Normal (63.2%) and U2R→Probe (45.4%) dominate errors.

4.2. Energy, Latency, and Memory

Table 4 presents benchmark results from ST Edge AI Developer Cloud on hosted STM32N6570-DK boards (remote benchmark service), rather than local emulation [20,21]. INT8 NPU execution yields **2.7**× speedup on NSL-KDD and **4.2**× on UNSW-NB15 over the same model on the Cortex-M55 CPU. At an estimated 150 mW (ST AN5946 [19]), this translates to 44–69 µJ per inference, i.e., **114–179**× lower than Chehade et al.'s 7.86 mJ on the STM32F7 [1]. UNSW-NB15 achieves 100% NPU execution (4 HW, 0 SW epochs); Flash shrinks 3.4–3.8× from FP32 to INT8. Both ReLU INT8 models on CICIDS2017 and IoT-23 achieve 100% NPU execution (Table 4), with latencies of 0.42 and 0.38 ms. The same sub-millisecond operating regime is therefore observed across all four datasets.

The energy figure is derived from the reference workload in AN5946, not direct measurement of our model. On-board validation with an STLINK-V3PWR probe is left to future work.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

Table 4. NPU benchmark on STM32N6570-DK. Energy estimated from AN5946 [19] (∼150 mW at nominal frequency); FP32 CPU-only power differs and is not estimated (—).

Model	Dataset	Time (ms)	Energy (µJ)	HW	Hyb	SW	Flash (KB)	RAM (KB)
ReLU FP32	NSL-KDD	1.24	—	0	0	11	466.4	2.17
ReLU INT8	NSL-KDD	0.46	69	5	1	2	137.7	1.25
ReLU FP32	UNSW-NB15	1.23	—	0	0	11	461.9	2.14
ReLU INT8	UNSW-NB15	**0.29**	**44**	4	0	0	120.6	0.50
ReLU INT8	CICIDS2017	0.42	63	4	0	0	120.6	0.50
ReLU INT8	IoT-23	0.38	57	4	0	0	105.0	0.50
QCFS INT8	NSL-KDD	0.54	81	13	1	14	138.0	2.00

4.3. INT8 Quantization Robustness

A 24-configuration ablation (3 calibration methods × 2 granularities × 4 sample sizes) shows that INT8 accuracy deviates from FP32 by at most 0.82 pp on NSL-KDD across all configurations. Several INT8 configurations slightly *improve* over FP32, likely due to regularization effects of quantization noise.

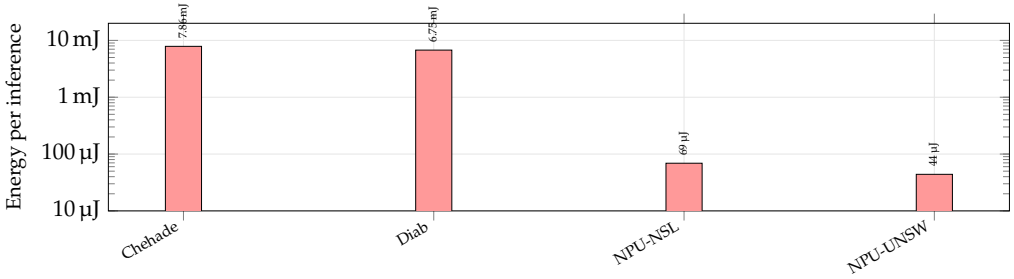

Figure 2. Energy per inference (log scale). Our NPU path achieves over two orders of magnitude reduction versus CPU-only MCU baselines. NPU = ReLU INT8 on Neural-ART (estimated from AN5946). Chehade [1]: STM32F7; Diab [2]: RPi 3B+.

Table 5 presents layer-wise activation comparison between FP32 and INT8 models on 1,000 NSL-KDD test samples. Hidden layers show moderate cosine similarity (0.65–0.68) due to per-neuron quantization noise, but the logit layer recovers to 0.978 and the overall prediction agreement is 99.0%. INT8 maps each activation to one of 256 discrete levels; the bounded per-neuron noise does not propagate into classification errors. From the T=1 SNN perspective, the quantized model (INT8) closely reproduces the firing decisions of the FP32 model.

Table 5. Layer-wise FP32 vs INT8 comparison (1,000 NSL-KDD samples).

Layer	Cosine Sim	MAE	L_∞
Relu_0 (256)	0.667	0.435	43.7
Relu_1 (256)	0.655	0.438	62.6
Relu_2 (128)	0.683	0.400	65.2
Logits (5)	**0.978**	0.103	19.8

4.4. QCFS Hyperparameter Sweep

The QCFS activation introduces a quantization-level hyperparameter L; prior work [3,4] uses $L \in \{4, 8, 16\}$ without systematic justification on IDS workloads. We sweep $L \in \{2, 4, 8, 16\}$ across 5 seeds on NSL-KDD and UNSW-NB15 (Table 6). On NSL-KDD all four L values are statistically indistinguishable (pairwise Wilcoxon $p \geq 0.81$), and on UNSW-NB15 L=4 is marginally preferred to L=2 (p=0.0625, which is the minimum attainable p-value for the signed-rank test at n=5, indicating an unambiguous direction but at the detection floor) but tied with L=8 and L=16 ($p \geq 0.31$). Since

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

every `Floor` triggers a CPU fallback and the operator count grows roughly linearly with L, L=4 is the Pareto-optimal choice: no accuracy gain from larger L, and half the CPU-fallback operator budget of L=8. To our knowledge this is the first empirical L-justification for QCFS in the IDS domain.

Table 6. QCFS L-sweep (mean±std over 5 seeds). Results refreshed from `results/qcfs_lsweep.json`.

Dataset	L=2	L=4	L=8	L=16
NSL-KDD OA	75.48±0.92	75.03±0.63	75.14±0.74	75.22±1.07
NSL-KDD MF1	52.40±1.59	51.17±0.83	50.94±1.45	51.07±1.86
UNSW-NB15 OA	64.85±0.20	65.76±0.19	65.59±0.14	65.56±0.54
UNSW-NB15 MF1	40.64±0.20	41.36±0.25	41.23±0.36	41.28±0.29

4.5. Statistical Significance

We test ReLU-MLP vs. each baseline (QCFS, TinyCNN, Random Forest) using a paired Wilcoxon signed-rank test on per-seed overall-accuracy differences, with Holm–Bonferroni family-wise error correction across the primary comparisons per dataset (Table 7). Effect sizes are reported as Cohen's d_z; 95% percentile-bootstrap confidence intervals are computed over 10,000 resamples. At n=5 (CICIDS2017) the smallest attainable two-sided p-value of the signed-rank test is 0.0625, so we treat CICIDS2017 tests as power-limited. The ReLU vs. QCFS comparison measures p=0.312, sitting well above the floor, i.e., a genuine non-rejection. The ReLU vs. TinyCNN comparison, by contrast, is exactly at the floor (p=0.063, all five seeds in the same direction, d_z=+1.32); we report it but do not claim significance at α=0.05. We therefore add paired TOST with a pre-specified ±1.0 pp bound (ReLU vs. QCFS, overall accuracy and macro-F1): UNSW-NB15 passes, while NSL-KDD/CICIDS2017/IoT-23 remain inconclusive under current seed counts.

Table 7. Paired Wilcoxon signed-rank tests on per-seed overall accuracy. p_{adj} is Holm–Bonferroni adjusted across all comparisons per dataset; reject at α=0.05. Cohen's d_z is the paired effect size (positive = row-one model higher). n is reported per row and reflects matched seeds between the two models; when arms have unequal seed counts the test uses the first n seeds of the longer arm.

Dataset	Comparison	p	p_{adj}	d_z	Reject?
NSL-KDD	ReLU vs. QCFS (n=20)	0.227	0.227	+0.26	✗
	ReLU vs. TinyCNN (n=10)	0.027	0.055	−0.97	✗
UNSW-NB15	ReLU vs. QCFS (n=10)	0.846	0.846	+0.19	✗
	ReLU vs. TinyCNN (n=10)	0.002	0.004	+2.06	✓
CICIDS2017	ReLU vs. QCFS (n=5)	0.312	0.312	+0.63	✗
	ReLU vs. TinyCNN (n=5)	0.063	0.125	+1.32	✗
IoT-23	ReLU vs. QCFS (n=5)	0.438	0.438	−0.33	✗

4.6. CICIDS2017 and IoT-23 Results

CICIDS2017 extends the evaluation from legacy and mid-decade benchmarks to a modern 15-class intrusion dataset. The ReLU MLP achieves 91.89±1.21 overall accuracy and 56.35±2.80 macro-F1, with ROC-AUC 0.992 and weighted-F1 94.0%. The QCFS L=4 arm reaches 90.99±0.48 / 57.65±2.38 (paired Wilcoxon p=0.312, n=5), sitting ≈0.9 pp below the ReLU mean on overall accuracy, consistent with the same T=1 approximation trend observed on NSL-KDD (p=0.227) and UNSW-NB15 (p=0.846). Per-class performance is bimodal: DDoS reaches F1 98.1% and benign 94.7%, while extreme-rare classes collapse (SQL Injection F1 1.4% on 4 test samples, Bot 6.3%, XSS 8.5%).

IoT-23 adds a contemporary IoT botnet dataset (5-class, 4.8 M training samples). The ReLU MLP achieves 75.59±2.71 overall accuracy and 66.41±1.50 macro-F1. Okiru is near-perfectly classified (F1>99.9%), while DDoS reaches 99.9% and C&C collapses to F1=5.3% due to severe underrepresentation. The QCFS arm reaches 77.65±1.18 / 67.40±0.52 (Wilcoxon p=0.438, n=5), extending the

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

same non-rejection pattern to all four datasets. These minority-class patterns mirror UNSW-NB15 and confirm that the deployment limit is the long-tail sample count, not the NPU.

4.7. Model Capacity Is Not the Bottleneck

A 10-seed Pareto sweep over four MLP sizes (logistic regression 210/350 p; MLP 64-64 7.3 K p; MLP 128-64 13.8/14.3 K p; our MLP 256-256-128 110/111 K p) shows that on NSL-KDD every architecture lands within 0.5 pp of the 256-256-128 baseline (78.40–78.86%), and logistic regression alone reaches 78.42%. On UNSW-NB15, a single 64-64 hidden layer already captures 64.52%, within 0.23 pp of the full model. The flat top of the curve means that, within our tested NPU-friendly MLP family, widening alone does not close the observed gap to tree ensembles. This is consistent with a data/feature-signal limitation, but is not a general proof that capacity is never the bottleneck. A 10-seed focal-loss ablation ($\gamma \in \{0, 1, 2, 3\}$) agrees: the best NSL-KDD configuration ($\gamma=2$) gains 0.52 pp overall and 1.32 pp macro-F1 over weighted cross-entropy, within the seed variance and therefore not claimed as a primary result.

4.8. When INT8 Equivalence Breaks Down

The 24-configuration quantization ablation (Sec. 4.3) produced a benign result on NSL-KDD (drop $\in [-0.82, +0.17]$ pp across all configurations), but on UNSW-NB15 the same sweep yields drops of +0.85 pp (best) to +11.82 pp (worst, percentile calibration), with a mean of +6.47 pp. The layer-wise FP32–INT8 cosine similarity is also much lower on UNSW (Relu_0: 0.32, Relu_1: 0.42, Relu_2: 0.57, logits: 0.91) than on NSL-KDD (0.67/0.65/0.68/0.98), and the UNSW FP32/INT8 prediction agreement falls to 75.2%, versus 99.0% on NSL-KDD. The proximate cause is that UNSW's FP32 margin between correct and incorrect classes is already small (macro-F1 40%), so quantization noise crosses more decision boundaries. **This marks a practical limit of the $T=1$ approximation:** it is strongest when the FP32 model is already confident, and can weaken as the margin shrinks. Practical deployers should calibrate with MinMax on a small ($\leq$ 100-sample) corpus, which empirically gives the best UNSW drop; and treat UNSW-scale margin loss as an expected cost, not a bug.

5. Discussion and Conclusions

Design-space positioning. This work targets a different point from GPU-based IDS: sub-millisecond, sub-millijoule inference on a commodity MCU. The absolute accuracy (78.6% NSL-KDD, 64.7% UNSW-NB15, 91.9% CICIDS2017, 75.6% IoT-23) is below GPU-simulation and binary-classification results, reflecting multi-class evaluation (5/10/15/5-class vs. binary) and NPU operator constraints (shallow MLP or Conv2D 1×3). Improving accuracy within these constraints is orthogonal to the deployment findings, and the statistically tested (Table 7) differences between ReLU MLP, TinyCNN, and QCFS remain small.

Why NPU over CPU? NPU inference frees the Cortex-M55 for RTOS tasks (packet capture, feature extraction), whereas CPU-only inference blocks the processor for 1.2 ms. Tree models achieve higher accuracy on UNSW-NB15, but in our tested toolchain/backend they have no Neural-ART execution path (`TreeEnsembleClassifier` rejected at compile time).

QCFS barrier. The `Floor` operator is absent from the Neural-ART operator set, forcing CPU fallback with 17.6% latency overhead. ReLU INT8 remains the optimal activation for commodity MCU NPUs.

MLP vs. TinyCNN. Ranking is dataset-dependent: TinyCNN trends higher on NSL-KDD ($d_z=-0.97$, $p_{\text{adj}}=0.055$), while MLP is superior on UNSW-NB15 ($p_{\text{adj}}=0.004$, $d_z=2.06$). A deployment can host both models (combined Flash <300 KB) and dispatch by traffic feature profile.

Limitations. *Energy is estimated, not measured*: we derive 44–69 µJ from AN5946's reference workload [19] rather than instrumenting our model on-silicon. Direct measurement with an STLINK-V3PWR probe remains future work. *Pipeline scope*: we benchmark classifier inference on pre-extracted flow features. Packet capture, flow reassembly, and feature extraction are not included in these timings. *Dataset coverage*: NSL-KDD is derived from 1998 DARPA traffic, CICIDS2017 has documented label-

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

alignment issues [16] while UNSW-NB15 and IoT-23 contain severe class imbalance. *Scope*: novelty is bounded by the Supplementary File S1 search window (April 2026), and architecture results are limited to MLP and Conv2D 1×3 TinyCNN under current Neural-ART operator support.

Conclusion. We demonstrate sub-millisecond classifier inference on a commodity MCU NPU: 2.7–4.2× CPU speedup and an estimated 114–179× energy reduction vs. recent STM32F7 deployments. Paired Wilcoxon tests show $T{=}1$ QCFS arms are statistically indistinguishable from INT8 ReLU across all four datasets (NSL-KDD $p{=}0.227$, UNSW $p{=}0.846$, CICIDS $p{=}0.312$, IoT-23 $p{=}0.438$), while TOST with ± 1.0 pp bound supports equivalence on UNSW and remains inconclusive on the other three datasets. Code and models will be released upon acceptance.

References

1. Chehade, A.; Ragusa, E.; Gastaldo, P.; Zunino, R. Energy-Efficient Deep Learning for Traffic Classification on Microcontrollers. In Proceedings of the Proc. IEEE ISCC, 2025. arXiv:2506.10851.
2. Diab, A.; Chehade, A.; Ragusa, E.; Gastaldo, P.; Zunino, R.; Baghdadi, A.; Rizk, M. Intrusion Detection on Resource-Constrained IoT Devices with Hardware-Aware ML and DL. *arXiv preprint arXiv:2512.02272* **2025**.
3. Bu, T.; Fang, W.; Ding, J.; Dai, P.; Yu, Z.; Huang, T. Optimal ANN-SNN Conversion for High-accuracy and Ultra-low-latency Spiking Neural Networks. In Proceedings of the Proc. ICLR, 2022. arXiv:2303.04347.
4. Jiang, H.; Anumasa, S.; De Masi, G.; Xiong, H.; Gu, B. A Unified Optimization Framework of ANN-SNN Conversion: Towards Optimal Mapping from Activation Values to Firing Rates. In Proceedings of the Proc. ICML, 2023, Vol. 202, *PMLR*.
5. Bu, T.; Li, M.; Yu, Z. Inference-Scale Complexity in ANN-SNN Conversion for High-Performance and Low-Power Applications. In Proceedings of the Proc. IEEE/CVF CVPR, 2025. arXiv:2409.03368.
6. Ngo, D.M.; Lightbody, D.; Temko, A.; Murphy, C.C.; Popovici, E. HH-NIDS: Hardware Heterogeneous Network Intrusion Detection System Using MAX78000. *Future Internet* **2022**, *15*, 9. https://doi.org/10.3390/fi15010009.
7. Zahm, W.; Nishibuchi, G.; Jose, A.; Chelian, S.; Vasan, S. Low-Power Cybersecurity Attack Detection Using Deep Learning on Neuromorphic Technologies. Technical report, CSIAC, 2024.
8. Farooq, M.A.; Rafique, A.; Fahmy, S.A.; Arora, A. High Throughput Low Latency Network Intrusion Detection on FPGAs: A Raw Packet Approach. In Proceedings of the Proc. IEEE IPDPSW, 2025, pp. 1201–1207. Edge-IIoT dataset, up to 1162M inferences/sec on Virtex Ultrascale+ FPGA, https://doi.org/10.1109/IPDPSW66978.2025.00192.
9. Wang, Z.; et al. An efficient intrusion detection model based on convolutional spiking neural network. *Scientific Reports* **2024**, *14*, 7054. https://doi.org/10.1038/s41598-024-57691-x.
10. Prajwalasimha, S.N.; et al. Event-Driven Intrusion Detection Systems using Spiking Neural Networks for Edge and IoT Security. In Proceedings of the Proc. IEEE ICSCSA, 2025. https://doi.org/10.1109/ICSCSA66339.2025.11171294.
11. Karthik, M.G.; et al. Energy-efficient intrusion detection with a protocol-aware transformer–spiking hybrid model. *Scientific Reports* **2026**, *16*, 7095. https://doi.org/10.1038/s41598-026-37367-4.
12. STMicroelectronics. *ST Neural-ART NPU Concepts*, 2025. UM3225.
13. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A Detailed Analysis of the KDD CUP 99 Data Set. In Proceedings of the Proc. IEEE CISDA, 2009, pp. 1–6. https://doi.org/10.1109/CISDA.2009.5356528.
14. Moustafa, N.; Slay, J. UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems. In Proceedings of the Proc. MilCIS, 2015, pp. 1–6. https://doi.org/10.1109/MilCIS.2015.7348942.
15. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proceedings of the Proc. ICISSP, 2018, pp. 108–116. https://doi.org/10.5220/0006639801080116.
16. Engelen, G.; Rimmer, V.; Joosen, W. Troubleshooting an Intrusion Detection Dataset: the CICIDS2017 Case Study. In Proceedings of the Proc. IEEE S&P Workshops (SPW), 2021, pp. 7–12. https://doi.org/10.1109/SPW53761.2021.00009.
17. Garcia, S.; Parmisano, A.; Erquiaga, M.J. IoT-23: A Labeled Dataset with Malicious and Benign IoT Network Traffic. *Zenodo Dataset* **2020**. Dataset landing page: https://www.stratosphereips.org/datasets-iot23, https://doi.org/10.5281/zenodo.4743746.
18. STMicroelectronics. *ONNX Toolbox Support*, 2025. ST Edge AI Developer Cloud Embedded Documentation.
19. STMicroelectronics. *How to Optimize Low-Power Modes on STM32N6 MCUs*, 2025. Application Note AN5946.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 April 2026 doi:10.20944/preprints202603.0817.v3

20. STMicroelectronics. *STEDGEAI-DC: Artificial Intelligence Developer Cloud for ST Microcontrollers, Neural-ART Accelerator, Microprocessors, and Smart Sensors*, 2025. Data Brief.
21. STMicroelectronics. *AI: Getting Started with ST Edge AI Developer Cloud*, 2025. STM32 Wiki.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

© 2026 by the author(s). Distributed under a Creative Commons CC BY license.